L Number	Hits	Search Text	DB	Time stamp
6	0	717/151,152,120	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/31 13:31
7	546	717/151,152,120.ccls.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/31 13:31
8	2465	hash with code	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/31 13:37
9	11	717/151,152,120.ccls. and (hash with code)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/31 13:32
10	4402	hash near value	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/31 13:37
11	10	(hash near value) and 717/151,152,120.ccls.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/31 13:37
-	1	("6308320").PN.	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2004/08/31 10:22
-	10	(("5204960") or ("5375242") or ("5655122") or ("5815720") or ("5850554") or ("5230050") or ("5301327") or ("5313387") or ("5325531") or ("5586328")).PN.	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2004/08/30 14:51
-	639	hash with header	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/30 14:51
-	323	717/162	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/30 14:54
-	226	(717/162).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/30 14:54
-	1	(hash with header) and ((717/162).CCLS.)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/30 14:56
-	1	("0000717").PN.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/30 14:56
-	6924	717.clas.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/30 14:56

-	11	717.clas. and (hash with header)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/30 14:56
-	10	("5204960" | "5230050" | "5301327" | "5313387" | "5325531" | "5375242" | "5586328" | "5655122" | "5815720" | "5850554").PN.	USPAT	2004/08/31 09:29
-	95	("6202203" "6035124" "5768596" "6131189" "5889995" "5960197" "5680622" "5768593" "6286135" "6149318" "5930510" "6026240" "5790867" "6064820" "6072952" "6117185" "6442663" "5768592" "5950003" "6059841" "6059841" "5812855" "5900001" "5903900" "5911144" "5915255" "5920876" "6038572" "6049810" "6115782" "4782444" "5202982" "5862385" "6077314" "6351849" "6401117" "6421730" "6625689" "6701338" "5748960" "6105072" "5187789" "5442790" "5937196" "5966536" "6090156" "5287511" "6151703" "5355469" "5524244").pn.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/08/31 10:22

-	49	("6202203" "6035124" "5768596" "6131189" "5889995" "5960197" "5680622" "5768593" "6286135" "6149318" "5930510" "6026240" "5790867" "6064820" "6072952" "6117185" "6442663" "5768592" "5950003" "6059841" "6059841" "5812855" "5900001" "5903900" "5911144" "5915255" "5920876" "6038572" "6049810" "6115782" "4782444" "5202982" "5862385" "6077314" "6351849" "6401117" "6421730" "6625689" "6701338" "5748960" "6105072" "5187789" "5442790" "5937196" "5966536" "6090156" "5287511" "6151703" "5355469" "5524244").pn.	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2004/08/31 10:23

Web

Results **1 - 10** of about **141** for **elfhash**. (0.24 seconds)

ElfHash c translation - PowerBASIC Forums
... unsigned long **ElfHash**(const unsigned char *key) { unsigned long h = 0; while (*key) {
h = (h << 4) + *key++; unsigned long g = h & 0xF0000000L; if (g) h ^= g ...
www.powerbasic.com/support/ forums/Forum6/HTML/001338.html - 18k - Cached - Similar pages

Anyone familiar with ELFHash algorithm?
... Anyone familiar with **ELFHash** algorithm? from Scott Johnson, [Bookmark Link].
To ... Subject: Anyone familiar with **ELFHash** algorithm? From ...
www.realsoftware.com/listarchives/ realbasic-nug/1999-11/msg00255.html - 8k - Cached - Similar pages

search.cpan.org: Digest::Elf - Perl extension for generating ...
... NAME ^. Digest::Elf - Perl extension for generating **ElfHash** values. SYNOPSIS ^. ...
DESCRIPTION ^. Digest::Elf is an implimentation of the **ElfHash** algorithm. ...
search.cpan.org/~mckay/Digest-Elf-1.4/Elf.pm - 4k - Cached - Similar pages

usr/local/bin/perl use ExtUtils::testlib; use Digest::Elf; for ...
#!/usr/local/bin/perl use ExtUtils::testlib; use Digest::Elf; for(
@ARGV) { print Digest::Elf::hash($_), "\n"; } exit(0);
search.cpan.org/src/MCKAY/Digest-Elf-1.4/bin/**elfhash** - 1k - Cached - Similar pages
[More results from search.cpan.org]

"Practical Algoritms for Programming", Binstock, A.; Rex, J. ...
71 */ #include <stdio.h> unsigned int **ElfHash** (const unsigned char *name) { unsigned
long h = 0, g; char str[32]; while (*name) { h = (h << 4) + *name++ ...
www.daylight.com/meetings/mug2000/Kappler/ElfHash.c - 1k - Cached - Similar pages

ELFHash.java -- Hashing function used in ELF-format object files ...
// === **ELFHash**.java -- Hashing function used in
ELF-format object files // Author : John ...
https://www.se.auckland.ac.nz/courses/ SOFTENG250/laboratories/Ensor/ELFHash.java - 2k - Cached - Similar pages

courses.cs.vt.edu/~cs2604/spring02/Projects/4/elfhash.cpp
Similar pages

Elf Hash - FoxPro Wiki
... **ElfHash** is a general purpose hash algorithm. A good article on hash algorithms
that covers **ElfHash** can be found at http://www.ddj ...
fox.wikis.com/wc.dll?Wiki~ElfHash~VFP - 7k - Cached - Similar pages

#include "prolog.h" #include <stdlib.h> #include <math.h> #include ...
... errors.h" #include "rpctypes.h" #include "brdcast.h" #include "servdef.h" #include
"toolkit.h" #include "netmex.h" unsigned long **ElfHash**(const unsigned char ...
www-mpy.desy.de/netmex/duvalkern/source.330/keyeqm.c - 7k - Cached - Similar pages

[PDF] l ¹ f " 63-¥ Û rTKz
File Format: PDF/Adobe Acrobat - View as HTML
... L Ÿß© ª ü ¸ ÓD Ïwï¥ 3 ¯1 Û rT"z ¥ **ELFhash** f" URL ¥Û rTi,z]H_ €
wi Õ URL Û rT "z ¥f"b 1oM hashing, **ELFhash**, URL, even ...
net.pku.edu.cn/~webg/twpaper/lxm_hashURL.pdf - Similar pages

Web   Images   Groups   News   Froogle   more »

Google™ | elfhash elf object | Search   Advanced Search
                                                                    Preferences

Web   Results **1 - 10** of about **30** for **elfhash elf object**. (0.35 seconds)

Elf Hash - FoxPro Wiki
... of **ElfHash**: *--- **ElfHash** ----- The published hash algorithm
used in the UNIX **ELF** format * for **object** files. ...
fox.wikis.com/wc.dll?Wiki~ElfHash~VFP - 7k - Cached - Similar pages

Classless.net :: Projects :: Classless.Hasher
... Cksum, 32, POSIX 1003.2-1992, CRC, 8, 16, 32, 64, See below, **ElfHash**, 32, UNIX **ELF**
format hash for **object** files, FCS16, 16, RFC 1331, Fast Frame Check Sequence, ...
classlesshasher.sourceforge.net/ - 17k - Cached - Similar pages

ELFHash.java -- Hashing function used in ELF-format object files ...
// === **ELFHash**.java -- Hashing function used in
ELF-format **object** files // Author : John ...
https://www.se.auckland.ac.nz/courses/ SOFTENG250/laboratories/Ensor/ELFHash.java - 2k - Cached - Similar pages

Linux Standard Base Core Specification 2.0rc3 Copyright © 2004 by ...
... Chapter 2. **Object** Files LSB-conforming implementations shall support the **object**
file Executable and Linking Format (**ELF**), which is defined by the following ...
www.linuxbase.org/spec/booksets/LSB-Core/LSB-Core.txt - 101k - Cached - Similar pages

1 Linux Standard Base Specification 1.2 2 3 Copyright © 2000 ...
... 784 785 Chapter 2. **Object** Files 786 787 LSB-conforming implementations shall support
the **object** file Executable and 788 Linking Format (**ELF**), which is defined ...
www.linuxbase.org/spec/ refspecs/LSB_1.2.0/gLSB_lines.txt - 101k - Cached - Similar pages
[More results from www.linuxbase.org]

The hash algorithm used in the UNIX ELF format for object files. ...
... used in the UNIX **ELF** format for **object** files. The input is a pointer to a string
to be hashed */ #define HASHSIZE 997 unsigned long **ElfHash**(const unsigned ...
remus.rutgers.edu/~rhoads/Code/hash1.c - 1k - Cached - Similar pages

[PDF] SWS ESPC 0003 B-02
File Format: PDF/Adobe Acrobat - View as HTML
... ARM **ELF** SWS ESPC 0003 B-02 Page 11 of 42 EI_MAG0 to to to to EI_MAG3—A file's first
4 bytes hold a magic number, identifying the file as an **ELF object** file ...
www.arm.com/pdfs/ARM%20ELF%20Specification.pdf - Similar pages

FreeLists / muscle / [muscle] Re: RFC: String class
... num, prec); +} + **ElfHash** ----- The published hash algorithm
used in the UNIX **ELF** format * for **object** files ...
www.freelists.org/archives/muscle/10-2003/msg00022.html - 11k - Cached - Similar pages

Nattyware / Papers / XUM32 and its implementation.
... hash function used in the UNIX **ELF** format for **object** files. Its ANSI C implementation
is here as it were defined in [3]: unsigned long **ElfHash** (const unsigned ...
www.nattyware.com/xum32txt.html - 8k - Aug 29, 2004 - Cached - Similar pages

Hash function in C
... used in the UNIX **ELF** format for **object** files. The input is a pointer to a string
to be hashed */ #define HASHSIZE 997 unsigned long **ElfHash**(const unsigned ...
forums.devshed.com/t54284/ s.html?highlight=Modulus+problem - 89k - Supplemental Result - Cached - Similar pages

P⊘RTAL

US Patent & Trademark Office

Search: ○ The ACM Digital Library   ⊙ The Guide

hash object optimization version

SEARCH

THE GUIDE TO COMPUTING LITERATURE

Feedback Report a problem Satisfaction survey

Terms used **hash object optimization version**   Found **69,135** of **826,312**

Sort results by	relevance	⬥ Save results to a Binder	Try an Advanced Search

Sort results by   relevance ▼   ⬥ Save results to a Binder   Try an Advanced Search
                                        ? Search Tips   Try this search in The Digital Library
Display results   expanded form ▼   ☐ Open results in a new window

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown   Relevance scale ☐ ⬜ ◨ ◧ ■

1 Data size optimizations for java programs
C. Scott Ananian, Martin Rinard
June 2003 **ACM SIGPLAN Notices , Proceedings of the 2003 ACM SIGPLAN conference on Language, compiler, and tool for embedded systems**, Volume 38 Issue 7

Full text available: 📄 pdf(349.36 KB)   Additional Information: full citation, abstract, references, citings, index terms

We present a set of techniques for reducing the memory consumption of object-oriented programs. These techniques include analysis algorithms and optimizations that use the results of these analyses to eliminate fields with constant values, reduce the sizes of fields based on the range of values that can appear in each field, and eliminate fields with common default values or usage patterns. We apply these optimizations both to fields declared by the programmer and to implicit fields in the runti ...

Keywords: bitwidth analysis, embedded systems, field externalization, field packing, size optimizations, static specialization

2 Optimizing object queries using an effective calculus
Leonidas Fegaras, David Maier
December 2000 **ACM Transactions on Database Systems (TODS)**, Volume 25 Issue 4

Full text available: 📄 pdf(641.65 KB)   Additional Information: full citation, abstract, references, citings, index terms, review

Object-oriented databases (OODBs) provide powerful data abstractions and modeling facilities, but they generally lack a suitable framework for query processing and optimization. The development of an effective query optimizer is one of the key factors for OODB systems to successfully compete with relational systems, as well as to meet the performance requirements of many nontraditional applications. We propose an effective framework with a solid theoretical basis for optimizing OODB query l ...

Keywords: nested relations, object-oriented databases, query decorrelation, query optimization

3 TID hash joins
Robert Marek, Erhard Rahm
November 1994 **Proceedings of the third international conference on Information and knowledge management**

Full text available: 📄 pdf(1.13 MB)   Additional Information: full citation, references, index terms

P⊘RTAL

US Patent & Trademark Office

Search: ○ The ACM Digital Library   ◉ The Guide

+hash value object header +optimization version +compiler

SEARCH

THE GUIDE TO COMPUTING LITERATURE

Feedback Report a problem Satisfaction survey

Terms used
hash value object header optimization version compiler

Found **1,463** of **826,312**

Sort results by | relevance ▾ | ◆ Save results to a Binder   Try an Advanced Search
Display results | expanded form ▾ | ? Search Tips   Try this search in The Digital Library
                                              ☐ Open results in a new window

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown   Relevance scale ☐ ▬ ▬ ▬ ■

1 **Data size optimizations for java programs**
C. Scott Ananian, Martin Rinard
June 2003 **ACM SIGPLAN Notices , Proceedings of the 2003 ACM SIGPLAN conference
on Language, compiler, and tool for embedded systems**, Volume 38 Issue 7

Full text available: 🗎 pdf(349.36 KB)   Additional Information: full citation, abstract, references, citings, index terms

We present a set of techniques for reducing the memory consumption of object-oriented
programs. These techniques include analysis algorithms and optimizations that use the
results of these analyses to eliminate fields with constant values, reduce the sizes of fields
based on the range of values that can appear in each field, and eliminate fields with
common default values or usage patterns. We apply these optimizations both to fields
declared by the programmer and to implicit fields in the runti ...

Keywords: bitwidth analysis, embedded systems, field externalization, field packing, size
optimizations, static specialization

2 **First International Workshop on Persistence and Java**
Malcolm Atkinson, Mick Jordan
November 1996 Technical Report, Sun Microsystems, Inc.

Full text available: 🗎 pdf(1.54 MB)   Additional Information: full citation, abstract

These proceedings record the First International Workshop on Persistence and Java, which
was held in Drymen, Scotland in September 1996. The focus of this workshop was the
relationship between the Java languages and long-term data storage, such as databases and
orthogonal persistence. There are many approaches being taken, some pragmatic and some
guided by design principles. If future application programmers building large and long-lived
systems are to be well-supported, it is essential that the ...

3 **An efficient implementation of SELF a dynamically-typed object-oriented language
based on prototypes**
C. Chambers, D. Ungar, E. Lee
September 1989 **ACM SIGPLAN Notices , Conference proceedings on Object-oriented
programming systems, languages and applications**, Volume 24 Issue 10

Full text available: 🗎 pdf(2.41 MB)   Additional Information: full citation, abstract, references, citings, index terms

We have developed and implemented techniques that double the performance of
dynamically-typed object-oriented languages. Our SELF implementation runs twice as fast
as the fastest Smalltalk implementation, despite SELF's lack of classes and explicit variables.

P◎RTAL

Subscribe (Full Service)   Register (Limited Service, Free)   Login

Search: ○ The ACM Digital Library   ◉ The Guide

`+hash +optimization +compiler`   **SEARCH**

THE GUIDE TO COMPUTING LITERATURE

♀ Feedback Report a problem Satisfaction survey

Terms used **hash** **optimization** **compiler**   Found **1,463** of 826,312

Sort results by   `relevance ▾`   ⬥ Save results to a Binder   Try an Advanced Search
                                       ❓ Search Tips   Try this search in The Digital Library
Display results   `expanded form ▾`   ☐ Open results in a new window

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10 · next
Best 200 shown   Relevance scale ☐ ▭ ▬ ■ ■

1 Compiler-directed run-time monitoring of program data access   ■
Chen Ding, Yutao Zhong
June 2002 **ACM SIGPLAN Notices , Proceedings of the workshop on Memory system performance**, Volume 38 Issue 2 supplement
Full text available: 🗎 pdf(1.40 MB)   Additional Information: full citation, abstract, references, citings

> Accurate run-time analysis has been expensive for complex programs, in part because most methods perform on all a data. Some applications require only partial reorganization. An example of this is off-loading infrequently used data from a mobile device. Complete monitoring is not necessary because not all accesses can reach the displaced data. To support partial monitoring, this paper presents a framework that includes a source-to-source C compiler and a run-time monitor. The compiler inserts ru ...

2 Control flow optimization for supercomputer scalar processing   ■
Pohua P. Chang, Wen-mei W. Hwu
June 1986 **Proceedings of the 3rd international conference on Supercomputing**
Full text available: 🗎 pdf(1.04 MB)   Additional Information: full citation, abstract, references, citings, index terms

> Control intensive scalar programs pose a very different challenge to highly pipelined supercomputers than vectorizable numeric applications. Function call/return and branch instructions disrupt the flow of instructions through the pipeline, degrading the utilization of the pipelined datapaths. This paper describes control flow optimization for scalar processing using an optimizing compiler. To obtain program control flow information, a system independent profiler has been integrated into th ...

3 Flick: a flexible, optimizing IDL compiler   ■
Eric Eide, Kevin Frei, Bryan Ford, Jay Lepreau, Gary Lindstrom
May 1997 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1997 conference on Programming language design and implementation**, Volume 32 Issue 5
Full text available: 🗎 pdf(1.75 MB)   Additional Information: full citation, abstract, references, citings, index terms

> An interface definition language (IDL) is a nontraditional language for describing interfaces between software components. IDL compilers generate "stubs" that provide separate communicating processes with the abstraction of local object invocation or procedure call. High-quality stub generation is essential for applications to benefit from component-based designs, whether the components reside on a single computer or on multiple networked hosts. Typical IDL compilers, ...

4 Look ma, no hashing, and no arrays neither   ■
Jiazhen Cai, Robert A. Paige